

breath and spirit of the present invention as disclosed herein.

CLAIMS

What is claimed is:

1 1. A method for controlling the transmission of data
2 packets through a network by controlling a Transmission
3 Control Protocol (TCP) rate in a network device having a
4 shared buffer with shared buffer space, the method
5 comprising:

6 organizing a forward data buffer into one or more
7 queues that store at least one forward data
8 packet;

9 calculating the network device's advertised window
10 size by implementing an integral control
11 algorithm that uses information pertaining to
12 the one or more queues;

13 providing the network device's advertised window
14 size to a TCP source; and

15 calculating a dynamic buffer threshold based, at
16 least in part, upon the sum of the queue sizes
17 and the shared buffer space.

18

1 2. The method of claim 1 wherein the step of organizing
2 a forward data buffer further comprises:

3 organizing the forward data buffer into one or more
4 queues with one queue per service class.

1 3. The method of claim 1 wherein the at least one
2 forward data packet is stored according to its service
3 class.

1 4. The method of claim 1 wherein the step of
2 calculating a network device's advertised window size
3 further comprises:

4 initializing a timer to a predetermined time
5 interval Δt , and an iteration counter to a
6 predetermined initial value n ;

7 sampling a current queue size $q_i(n)$ during the
8 predetermined time interval Δt ;

9 calculating a current error signal $e_i(n)$ based, at
10 least in part, upon the current queue size
11 $q_i(n)$;

12 calculating the network device's advertised window
13 size $W_i(n)$, based, at least in part, upon the
14 current error signal $e_i(n)$ according to the
15 equation: $W_i(n) = [W_i(n-1) + \alpha e_i(n)]_{W_{\min}}^{W_{\max}}$, where α , W_{\max} ,
16 and W_{\min} , are predetermined parameters;

17 resetting the timer, upon expiration of the
18 predetermined interval Δt ; and

19 iterating the iteration counter, upon expiration of
20 the predetermined time interval Δt .

1 5. The method of claim 4 wherein the steps of
2 calculating a current error signal $e_i(n)$ and calculating

3 the network device's advertised window size further
4 comprise:

5 filtering the current error signal $e_i(n)$ according to
6 the relation: $\hat{e}_i(n) = (1 - \beta)\hat{e}_i(n-1) + \beta e_i(n)$, where β
7 is a predetermined parameter; and
8 calculating the network device's advertised window
9 size $W_i(n)$, based, at least in part, upon the
10 filtered current error signal $\hat{e}_i(n)$ according to
11 the equation: $W_i(n) = [W_i(n-1) + \alpha \hat{e}_i(n)]_{W_{\min}^{W_{\max}}}$, where α ,
12 W_{\max} , and W_{\min} , are predetermined parameters.

1 6. The method of claim 1 wherein the step of providing
2 the network device's advertised window size to a TCP
3 source further comprises:

4 carrying information relating to the network
5 device's advertised window size by returning
6 TCP acknowledgements in a receiver's advertised
7 window field.

1 7. The method of claim 1 wherein the step of providing
2 the network device's advertised window size to a TCP
3 source further comprises:

4 updating a TCP receiver's advertised window size.

1 8. The method of claim 7 wherein the step of updating a
2 TCP receiver's advertised window size further comprises:

3 identifying whether a packet is an ACK packet, and,
4 if not, putting the non-ACK packet in a reverse

5 data buffer;
6 determining a service class for the identified ACK
7 packet;
8 reading the TCP receiver's advertised window size
9 (RW_{rec}) and a checksum ($RCHKSUM$) from the
10 identified ACK packet;
11 determining whether the TCP receiver's advertised
12 window size RW_{REC} , is less than or equal to the
13 calculated network device's advertised window
14 size $W_i(n)$ and, if not setting a advertised
15 window field in the identified ACK packet equal
16 to the network device's advertised window size
17 $W_i(n)$ and updating the checksum field for the
18 identified ACK packet.

1 9. The method of claim 1 wherein the step of
2 calculating a dynamic buffer threshold further comprises:
3 initializing a timer to a predetermined time
4 interval Δs and an iteration counter to a
5 predetermined initial value n ;
6 setting an initial dynamic buffer threshold $T(0)$
7 equal to a parameter γ multiplied by a buffer
8 size B and divided by a number of service
9 classes K ;
10 sampling a current queue size $q_i(n)$ during the
11 predetermined time interval Δs ;
12 calculating a sum of the sampled current queue size

13 according to the equation: $Q(n) = \sum_{i=1}^K q_i(n)$;
14 determining whether the sum of the sampled current
15 queue size is less than the product of the
16 parameter and the buffer size γB ;
17 if so, updating the dynamic buffer threshold
18 according to $\min\{T(n-1) + \Delta T, \gamma B\}$, where ΔT is
19 a step size that controls the rate at
20 which the dynamic buffer threshold
21 changes;
22 if not, updating the dynamic buffer threshold
23 according to $\max\{T(n-1) - \Delta T, T_{\min}\}$, where T_{\min}
24 is a predetermined minimum size for the
25 dynamic buffer threshold;
26 resetting the timer, upon expiration of the
27 predetermined interval Δs ; and
28 iterating the iteration counter, upon expiration of
29 the predetermined time interval Δs .

1 10. The method of claim 9 wherein the step of
2 calculating a sum of the sampled current queue size
3 further comprises:

4 filtering the sum of the sampled current queue size
5 $Q(n)$ according to the relation:
6 $\hat{Q}(n) = (1 - \varphi)\hat{Q}(n-1) + \varphi Q(n)$, wherein φ is a
7 predetermined parameter.

1 11. An apparatus for controlling the transmission of
2 data packets through a network by controlling a
3 Transmission Control Protocol (TCP) rate in a network
4 device having a shared buffer with shared buffer space,
5 the apparatus comprising:

6 a forward data buffer, organized into one or more
7 queues that store at least one forward data
8 packet;

9 a network device's advertised window size
10 calculation module that calculates a network
11 device's advertised window size by implementing
12 an integral control algorithm that uses
13 information pertaining to the one or more
14 queues;

15 a feed back module that provides the network
16 device's advertised window size to a TCP
17 source; and

18 a dynamic buffer threshold module that calculates a
19 dynamic buffer threshold based, at least in
20 part, upon the sum of the queue sizes and the
21 shared buffer space.

1 12. The apparatus of claim 11 wherein the network
2 device's advertised window size calculation module
3 further comprises:

4 a timer, initially set to a predetermined time
5 interval Δt , and an iteration counter initially
6 set to a predetermined initial value n ;

7 a current queue size sampler that samples a current
8 queue size $q_i(n)$ during the predetermined time
9 interval Δt ;
10 a current error signal calculation module that
11 calculates a current error signal $e_i(n)$ based,
12 at least in part, upon the current queue size
13 $q_i(n)$;
14 a window size calculation module that calculates the
15 network device's advertised window size $W_i(n)$,
16 based, at least in part, upon the current error
17 signal $e_i(n)$ according to the equation:
18 $W_i(n) = [W_i(n-1) + \alpha e_i(n)]_{W_{\min}}^{W_{\max}}$, where α , W_{\max} , and W_{\min} ,
19 are predetermined parameters.

1 13. The apparatus of claim 12 wherein the current error
2 signal calculation module further comprises:
3 a filter module that filters the current error
4 signal $e_i(n)$ according to the relation:
5 $\hat{e}_i(n) = (1 - \beta)\hat{e}_i(n-1) + \beta e_i(n)$, where β is a
6 predetermined parameter; and
7 wherein the window size calculation module
8 calculates the network device's advertised
9 window size $W_i(n)$, based, at least in part, upon
10 the filtered current error signal $\hat{e}_i(n)$
11 according to the equation:
12 $W_i(n) = [W_i(n-1) + \alpha \hat{e}_i(n)]_{W_{\min}}^{W_{\max}}$, where α , W_{\max} , and W_{\min} ,
13 are predetermined parameters.

1 14. The apparatus of claim 11 wherein the feed back
2 module further comprises:

3 an advertised window size updating module that
4 updates a TCP receiver's advertised window
5 size.

1 15. The apparatus of claim 14 wherein the advertised
2 window size updating module further comprises:

3 an ACK packet identification module that identifies
4 whether a packet is an ACK packet, and, if not,
5 puts the non-ACK packet in a reverse data
6 buffer;

7 an ACK packet classifier that determines a service
8 class for the identified ACK packet;

9 an advertised window size reader that reads a TCP
10 receiver's advertised window size (RW_{rec}) and a
11 checksum ($RCHKSUM$) from the identified ACK
12 packet;

13 a window size comparison module that determines
14 whether the TCP receiver's advertised window
15 size RW_{REC} , is less than or equal to the
16 calculated network device's advertised window
17 size $W_i(n)$ and, if not sets an advertised window
18 field in the identified ACK packet equal to the
19 calculated network device's advertised window
20 size $W_i(n)$ and updates the checksum field for
21 the identified ACK packet.

1 16. The apparatus of claim 11 wherein the dynamic buffer
2 threshold module further comprises:

3 a timer initially set to a predetermined time
4 interval Δs and an iteration counter initially
5 set to a predetermined initial value n ;

6 a current queue size sampler that samples a current
7 queue size $q_i(n)$ during the predetermined time
8 interval Δs ;

9 a current queue size calculation module that
10 calculates a sum of the sampled current queue
11 size according to the equation: $Q(n) = \sum_{i=1}^K q_i(n)$,

12 where K is a number of service classes;

13 a dynamic buffer threshold determiner that
14 determines whether the sum of the sampled
15 current queue size is less than the product of
16 a parameter γ and a buffer size B ;

17 and an updating module that updates the dynamic
18 buffer threshold if the sum of the sampled
19 current queue size is less than the product of
20 the parameter γ and the buffer size B ,
21 according to $\min\{T(n-1) + \Delta T, \gamma B\}$, where ΔT is a
22 step size that controls the rate at which the
23 dynamic buffer threshold changes and if the sum
24 of the sampled current queue size is not less
25 than the product of the parameter γ and a
26 buffer size B , updates the dynamic buffer

27 threshold according to $\max\{T(n-1)-\Delta T, T_{\min}\}$, where
28 T_{\min} is a predetermined minimum size for the
29 dynamic buffer threshold.

1 17. The apparatus of claim 16 wherein the current queue
2 size calculation module further comprises:

3 a filter that filters the sum of the sampled current
4 queue size $Q(n)$ according to the relation:

5 $\hat{Q}(n) = (1-\varphi)\hat{Q}(n-1) + \varphi Q(n)$, wherein φ is a
6 predetermined parameter.

1 18. An article of manufacture for controlling the
2 transmission of data packets through a network by
3 controlling a Transmission Control Protocol (TCP) rate in
4 a network device having a shared buffer with shared
5 buffer space, the article of manufacture comprising:

6 at least one processor readable carrier; and
7 instructions carried on the at least one carrier;
8 wherein the instructions are configured to be
9 readable from the at least one carrier by at least one
10 processor and thereby cause the at least one processor to
11 operate so as to:

12 organize a forward data buffer into one or more
13 queues that store at least one forward data
14 packet;

15 calculate a network device's advertised window size
16 by implementing an integral control algorithm
17 that uses information pertaining to the one or

18 more queues;
19 provide the network device's advertised window size
20 to a TCP source; and
21 calculate a dynamic buffer threshold based, at least
22 in part, upon the sum of the queue sizes and
23 the shared buffer space.

1 19. The article of manufacture of claim 18 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:

5 organize the forward data buffer into one or more
6 queues with one queue per service class.

1 20. The article of manufacture of claim 18 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:

5 store the at least one forward data packet according
6 to its service class.

1 21. The article of manufacture of claim 18 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:

5 initialize a timer to a predetermined time interval
6 Δt , and an iteration counter to a predetermined
7 initial value n ;

8 sample a current queue size $q_i(n)$ during the
9 predetermined time interval Δt ;
10 calculate a current error signal $e_i(n)$ based, at
11 least in part, upon the current queue size
12 $q_i(n)$;
13 calculate the network device's advertised window
14 size $W_i(n)$, based, at least in part, upon the
15 current error signal $e_i(n)$ according to the
16 equation: $W_i(n) = [W_i(n-1) + \alpha e_i(n)]_{W_{\min}}^{W_{\max}}$, where α , W_{\max} ,
17 and W_{\min} , are predetermined parameters;
18 reset the timer, upon expiration of the
19 predetermined interval Δt ; and
20 iterate the iteration counter, upon expiration of
21 the predetermined time interval Δt .

1 22. The article of manufacture of claim 21 wherein
2 the instructions are configured to be readable from the
3 at least one carrier by at least one processor and
4 thereby cause the at least one processor to operate so as
5 to:

6 filter the current error signal $e_i(n)$ according to
7 the relation: $\hat{e}_i(n) = (1 - \beta)\hat{e}_i(n-1) + \beta e_i(n)$, where β
8 is a predetermined parameter; and
9 calculate the network device's advertised window
10 size $W_i(n)$, based, at least in part, upon the
11 filtered current error signal $\hat{e}_i(n)$ according to

12 the equation: $W_i(n) = [W_i(n-1) + \alpha \hat{e}_i(n)]_{W_{\min}^{W_{\max}}}$, where α ,
13 W_{\max} , and W_{\min} , are predetermined parameters.

1 23. The article of manufacture of claim 18 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:
5 carry information relating to the network device's
6 advertised window size by returning TCP
7 acknowledgements in a receiver's advertised
8 window field.

1 24. The article of manufacture of claim 18 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:
5 update a TCP receiver's advertised window size.

1 25. The article of manufacture of claim 24 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:
5 identify whether a packet is an ACK packet, and, if
6 not, put the non-ACK packet in a reverse data
7 buffer;
8 determine a service class for the identified ACK
9 packet;
10 read a TCP receiver's advertised window size (RW_{rec})

11 and a checksum (*RCHKSUM*) from the identified
12 ACK packet;
13 determine whether the TCP receiver's advertised
14 window size RW_{REC} , is less than or equal to the
15 calculated network device's advertised window
16 size $W_i(n)$ and, if not setting an advertised
17 window field in the identified ACK packet equal
18 to the calculated network device's advertised
19 window size $W_i(n)$ and updating the checksum
20 field for the identified ACK packet.

1 26. The article of manufacture of claim 18 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:
5 initialize a timer to a predetermined time interval
6 Δs and an iteration counter to a predetermined
7 initial value n ;
8 set an initial dynamic buffer threshold $T(0)$ equal to
9 a parameter γ multiplied by a buffer size B
10 and divided by a number of service classes K ;
11 sample a current queue size $q_i(n)$ during the
12 predetermined time interval Δs ;
13 calculate a sum of the sampled current queue size
14 according to the equation: $Q(n) = \sum_{i=1}^K q_i(n)$;
15 determine whether the sum of the sampled current
16 queue size is less than the product of the

17 parameter and the buffer size γB ;
18 if so, updating the dynamic buffer threshold
19 according to $\min\{T(n-1)+\Delta T, \gamma B\}$, where ΔT is
20 a step size that controls the rate at
21 which the dynamic buffer threshold
22 changes;
23 if not, updating the dynamic buffer threshold
24 according to $\max\{T(n-1)-\Delta T, T_{\min}\}$, where T_{\min}
25 is a predetermined minimum size for the
26 dynamic buffer threshold;
27 reset the timer, upon expiration of the
28 predetermined interval Δs ; and
29 iterate the iteration counter, upon expiration of
30 the predetermined time interval Δs .

1 27. The article of manufacture of claim 26 wherein the
2 instructions are configured to be readable from the at
3 least one carrier by at least one processor and thereby
4 cause the at least one processor to operate so as to:

5 filter the sum of the sampled current queue size
6 $Q(n)$ according to the relation:
7 $\hat{Q}(n) = (1-\phi)\hat{Q}(n-1) + \phi Q(n)$, wherein ϕ is a
8 predetermined parameter.

1 28. A signal embodied in a carrier wave and representing
2 sequences of instructions which, when executed by at
3 least one processor, cause the at least one processor to

4 control the transmission of data packets through a
5 network by controlling a Transmission Control Protocol
6 (TCP) rate in a network device having a shared buffer
7 with shared buffer space, by performing the steps of:

8 organizing a forward data buffer into one or more
9 queues that store at least one forward data
10 packet;

11 calculating a network device's advertised window
12 size by implementing an integral control
13 algorithm that uses information pertaining to
14 the one or more queues;

15 providing the network device's advertised window
16 size to a TCP source; and

17 calculating a dynamic buffer threshold based, at
18 least in part, upon the sum of queue sizes and
19 the shared buffer space.

1 29. The signal of claim 28 wherein the step of
2 organizing a forward data buffer further comprises:

3 organizing the forward data buffer into one or more
4 queues with one queue per service class.

1 30. The signal of claim 28 wherein the at least one
2 forward data packet is stored according to its service
3 class.

1 31. The signal of claim 28 wherein the step of
2 calculating a network device's advertised window size
3 further comprises:

4 initializing a timer to a predetermined time
5 interval Δt , and an iteration counter to a
6 predetermined initial value n ;

7 sampling a current queue size $q_i(n)$ during the
8 predetermined time interval Δt ;

9 calculating a current error signal $e_i(n)$ based, at
10 least in part, upon the current queue size
11 $q_i(n)$;

12 calculating the network device's advertised window
13 size $W_i(n)$, based, at least in part, upon the
14 current error signal $e_i(n)$ according to the
15 equation: $W_i(n) = [W_i(n-1) + \alpha e_i(n)]_{W_{\min}}^{W_{\max}}$, where α , W_{\max} ,
16 and W_{\min} , are predetermined parameters;

17 resetting the timer, upon expiration of the
18 predetermined interval Δt ; and

19 iterating the iteration counter, upon expiration of
20 the predetermined time interval Δt .

1 32. The signal of claim 31 wherein the steps of
2 calculating a filtered current error signal $e_i(n)$ and
3 calculating the network device's advertised window size
4 further comprise:

5 filtering the current error signal $e_i(n)$ according to
6 the relation: $\hat{e}_i(n) = (1 - \beta)\hat{e}_i(n-1) + \beta e_i(n)$, where β

7 is a predetermined parameter; and
8 calculating the network device's advertised window
9 size $W_i(n)$, based, at least in part, upon the
10 filtered current error signal $\hat{e}_i(n)$ according to
11 the equation: $W_i(n) = [W_i(n-1) + \alpha \hat{e}_i(n)]_{W_{\min}}^{W_{\max}}$, where α ,
12 W_{\max} , and W_{\min} , are predetermined parameters.

1 33. The signal of claim 28 wherein the step of providing
2 the network device's advertised window size to a TCP
3 source further comprises:
4 carrying information relating to the network
5 device's advertised window size by returning
6 TCP acknowledgements in a receiver's advertised
7 window field.

1 34. The signal of claim 28 wherein the step of providing
2 the network device's advertised window size to a TCP
3 source further comprises:
4 updating a TCP receiver's advertised window size.

1 35. The signal of claim 34 wherein the step of updating
2 a TCP receiver's advertised window size further
3 comprises:
4 identifying whether a packet is an ACK packet, and,
5 if not, putting the non-ACK packet in a reverse
6 data buffer;
7 determining a service class for the identified ACK
8 packet;

9 reading a TCP receiver's advertised window size
10 (RW_{rec}) and a checksum ($RCHKSUM$) from the
11 identified ACK packet;
12 determining whether the TCP receiver's advertised
13 window size RW_{REC} , is less than or equal to the
14 calculated network device's advertised window
15 size $W_i(n)$ and, if not setting an advertised
16 window field in the identified ACK packet equal
17 to the calculated network device's advertised
18 window size $W_i(n)$ and updating the checksum
19 field for the identified ACK packet.

1 36. The signal of claim 28 wherein the step of
2 calculating a dynamic buffer threshold further comprises:
3 initializing a timer to a predetermined time
4 interval Δs and an iteration counter to a
5 predetermined initial value n ;
6 setting an initial dynamic buffer threshold $T(0)$
7 equal to a gain constant γ multiplied by a
8 buffer size B and divided by a number of
9 service classes K ;
10 sampling a current queue size $q_i(n)$ during the
11 predetermined time interval Δs ;
12 calculating a sum of the sampled current queue size
13 according to the equation: $Q(n) = \sum_{i=1}^K q_i(n)$;
14 determining whether the sum of the sampled current
15 queue size is less than the product of the gain

16 constant and the buffer size γB ;
17 if so, updating the dynamic buffer threshold
18 according to $\min\{T(n-1)+\Delta T, \gamma B\}$, where ΔT is
19 a step size that controls the rate at
20 which the dynamic buffer threshold
21 changes;
22 if not, updating the dynamic buffer threshold
23 according to $\max\{T(n-1)-\Delta T, T_{\min}\}$, where T_{\min}
24 is a predetermined minimum size for the
25 dynamic buffer threshold;
26 resetting the timer, upon expiration of the
27 predetermined interval Δs ; and
28 iterating the iteration counter, upon expiration of
29 the predetermined time interval Δs .

1 37. The signal of claim 36 wherein the step of
2 calculating a sum of the sampled current queue size
3 further comprises:

4 filtering the sum of the sampled current queue size
5 $Q(n)$ according to the relation:
6 $\hat{Q}(n) = (1-\phi)\hat{Q}(n-1) + \phi Q(n)$, wherein ϕ is a
7 predetermined parameter.

1 38. A computer data signal embodied in a carrier wave
2 readable by a computing system and encoding a computer
3 program of instructions for executing a computer process
4 performing the method recited in claim 1.